

Vitor Amadeu Souza

Introdução ao

React Native

No ambiente Expo Snack

Parte XVI

© 2024 by Cerne Tecnologia e Treinamento Ltda.

© 2024 by Vitor Amadeu Souza

Nenhuma parte desta publicação poderá ser reproduzida sem autorização prévia e escrita de **Cerne Tecnologia e Treinamento Ltda.** Este livro publica nomes comerciais e marcas registradas de produtos pertencentes a diversas companhias. O editor utiliza as marcas somente para fins editoriais e em benefício dos proprietários das marcas, sem nenhuma intenção de atingir seus direitos.

Outubro de 2024

Direitos reservados por:

Cerne Tecnologia e Treinamento Ltda

Produção: Cerne Tecnologia e Treinamento

E-mail da Empresa: cerne@cerne-tec.com.br

Home Page: www.cerne-tec.com.br.com.br

Atendimento ao Consumidor: sac@cerne-tec.com.br

Contato com o Autor: vitor@cerne-tec.com.br



FEITO NO BRASIL

***“Venha também sobre mim a tua benignidade, ó Senhor, e a tua
salvação, segundo a tua palavra.”***

Sl 119:41

Cerne Tecnologia

A Cerne Tecnologia tem uma equipe preparada para desenvolvimento de projetos eletrônicos em diversas áreas: Médica, Entretenimento, Industrial, Robótica, Científica, Automobilística, Aeronáutica, etc. Trabalhamos com tecnologia microcontrolada usando o PIC, ARM, AVR, 8051, dsPIC, PIC24, PIC32 além do Arduino, Raspberry, Beaglebone etc. Desenvolvemos o projeto desde sua concepção até a entrega do produto final, passando pelas etapas de esquema elétrico, protótipo e desenvolvimento de circuito impresso.

Desenvolvemos aplicativos para smartphones/tablets Android, iOS, Blackberry, Windows Phone e no desenvolvimento de softwares a nível PC para plataforma Windows, usando ferramentas como o Visual Basic, C# e C++.

Atuamos na parte de montagem de placas, onde podemos fornecer ambos os serviços de desenvolvimento de projetos e produção ou apenas um destes.

Desenvolvemos esquemas elétricos e layout de PCI, tanto em tecnologia convencional como SMD.

Temos a flexibilidade de customizar um de nossos produtos, de modo a atender a uma necessidade específica do cliente, tornando o custo de desenvolvimento menor se comparado a construção de um projeto desde a sua fase inicial.

Desenvolvemos e fornecemos kits didáticos para diversos microcontroladores além de apostilas, livros e e-books.

Na hora de desenvolver um projeto ou equipar seu laboratório não hesite em nos contatar. Entre em contato conosco através do endereço cerne-tec.com.br para obter mais informações.



Sumário

Capítulo I – Metodologia de desenvolvimento.....	6
1. Introdução.....	6
Capítulo II – Programação em React Native.....	7
1. Cores.....	7
2. Imagens aleatórias.....	9
3. Jogo da velha.....	11
4. Jogo da forca.....	15

Capítulo I

Metodologia de desenvolvimento

1. Introdução

Neste livro, você encontrará uma parte da série dedicada ao desenvolvimento de projetos com React Native. Ao longo desta obra, diversos experimentos serão apresentados conforme o sumário, permitindo um aperfeiçoamento contínuo no uso deste framework para a construção de interfaces responsivas com foco em dispositivos móveis.

Capítulo II

Programação em React Native

1. Cores

O código implementa um aplicativo em React Native chamado ColorApp, que exibe uma lista de cores. Dentro do componente, há um array de cores, contendo os valores "red", "blue", "green" e "yellow". O map é utilizado para iterar sobre esse array, e para cada cor, um componente View é criado com um fundo da respectiva cor e uma altura de 100 pixels. Dentro de cada View, há um componente Text que exibe o nome da cor. O layout é ajustado com flex: 1 para ocupar todo o espaço disponível na tela.

```
import React from 'react';
import { View, Text, Button } from 'react-native';

const ColorApp = () => {
  const colors = ['red', 'blue', 'green', 'yellow' ];

  return (
    <View style={{ flex: 1 }}>
      {colors.map((color) => (
        <View
          key={color}
          style={{ backgroundColor: color, height: 100 }}>
          <Text>{color}</Text>
        </View>
      ))}
    </View>
  )
}
```

```
    );  
};  
  
export default ColorApp;
```



2. Imagens aleatórias

O código implementa um aplicativo chamado ImageGalleryApp utilizando React Native, que exibe uma galeria de 100 imagens geradas aleatoriamente. O componente principal utiliza o ScrollView com um estilo de contêiner que centraliza e organiza as imagens em várias linhas, permitindo o scroll vertical. A função map cria um array de 100 elementos e, para cada um, gera um componente Image que obtém uma imagem aleatória do serviço picsum.photos. Cada imagem tem 100x100 pixels de dimensão e um espaçamento de 5 pixels entre elas. O layout usa flexDirection: 'row' e flexWrap: 'wrap' para alinhar as imagens em múltiplas linhas ajustadas na tela.

```
import React from 'react';
import { View, Image, ScrollView, StyleSheet } from 'react-native';

const ImageGalleryApp = () => {
  return (
    <ScrollView contentContainerStyle={styles.container}>
      {[...Array(100)].map((_, index) => (
        <Image
          key={index}
          source={{ uri: `https://picsum.photos/200?random=${
            index}` }}
          style={styles.image}
        />
      ))}
    </ScrollView>
  );
};

const styles = StyleSheet.create({
```

```

container: {
  marginTop: 50,
  alignItems: 'center',
  justifyContent: 'center',
  flexDirection: 'row',
  flexWrap: 'wrap',
},
image: {
  width: 100,
  height: 100,
  margin: 5, // Adiciona um pouco de espaço entre as imagens
},
});export default ImageGalleryApp;

```

